

УДК 004.021

**ПРОЕКТИРОВАНИЕ API И БАЗЫ ДАННЫХ ДЛЯ ВЕБ-ПРИЛОЖЕНИЯ,
РАЗРАБОТАННОГО ДЛЯ ПОДБОРА СОТРУДНИКОВ ПО НАВЫКАМ****П.В. МАХАНЬКО***(Представлено: Ю.Н. КРАВЧЕНКО)*

В данной статье рассматриваются цели и принципы моделирования API и базы данных для веб-приложения

Введение. В наше время трудно представить хорошее веб-приложение без разработанной базы данных и быстрого, оптимизированного API. Большинство современных веб приложений зачастую пользуются сторонними сервисами, которые позволяют оптимизировать и улучшить разработку. Базы данных позволяют хранить большие объемы данных, облачные хранилища позволяют доставлять контент к конечному пользователю независимо от его местоположения, а быстрые API позволяют обрабатывать запросы в кратчайшие сроки. API представляет собой промежуточное звено между клиентской частью приложения и базой данных. Так же API является отличным способом защиты пользовательских данных используя современные технологии OAuth и JSON Web Token.

Основной раздел. JSON Web token является открытым стандартом, создания токенов доступа основываясь на json. В момент авторизации пользователя, сервер генерирует хеш-ключи, которые отправляет клиентской части веб-приложения. В дальнейшем, каждый запрос к серверу, должен содержать в себе хеш-ключ для подтверждения прав доступа к ресурсам.

OAuth является современным решением защиты пользователей, а также удобным способом регистрировать и авторизировать пользователей в приложении. OAuth – это открытый протокол авторизации, который позволяет предоставить третьей стороне доступ к защищенным данным не передавая логин и пароль что позволяет пользователю иметь одну учетную запись для многих сервисов.

Преимущество OAuth является то, что пользователь хранит свои логин и пароль только в одном сервисе, например, Google, а для доступа к другим, он получает токен, который генерируется на стороне авторизации.

Для взаимодействия с почтовыми сервисами использовались открытые API Google и Microsoft Exchange. Данные сервисы позволяют обмениваться электронными письмами между различными адресами, а также при получении доступа к письмам третьей стороной, позволяет обрабатывать, анализировать и перенаправлять эти письма на другие адреса. Доступ получается при помощи открытых и секретных ключей, генерируемые на стороне этих сервисов.

Amazon S3 является отличным средством для хранения и получения больших объемов данных, таких как крупные файлы, документы, аудиозаписи и фильмы. Преимущество Amazon S3 является быстрая работа, простота интеграции в приложение, а также защищенность файлов при помощи шифрования.

Amazon EC2 – веб-сервис, который предоставляет вычислительные мощности в облаке, которые позволяют содержать ваше приложение, библиотеки и данные. Данный сервис отлично подходит для развертывания API и базы данных.

Amazon Cloudfront – цель данного сервиса, является доставка контента конечным пользователям с минимальными задержками и высокой скоростью передачи. При загрузке файла или приложений, Amazon самостоятельно создает реплики данных и хранит их на всех своих сервера. Это дает возможность получать данные разными пользователями из разных точек мира с одинаковой скоростью.

В качестве точек для запросов, сервер предоставляет веб-приложению конечные адреса, обращаясь к которым, веб-приложение получает определенные данные. Запросы проходят по стандартному HTTP протоколу, используя методы GET, POST. Чат и обмен файлами реализованы по технологии сокетов (протокол WebSocket). Сокет – протокол дуплексной связи между клиентом и сервером в режиме реального времени. В данном случае, веб-приложение подписывается на новые сообщения при успешном входе в приложение. Затем при получении нового сообщения оно мгновенно будет отправлено пользователю.

В качестве базы данных была выбрана документно-ориентированная база данных MongoDB. Синтаксис MongoDB схож с форматом json, а это в свою очередь не задает жестких рамок хранения данных, что позволяет удобно расширять хранимые данные. Вся база данных состоит из коллекций, а хранимые в ней данные – документы. Документ представляет из себя json-объект в виде пар ключ/значение, где в качестве значения могут храниться строки, числа, двоичные данные, массивы или другие вложенные объекты.

Реляционная модель представляет собой базу данных в виде множества взаимосвязанных отношений (таблиц), которые используются для хранения информации об объектах, представленных в базе данных.

В соответствии с требованиями программы, определены коллекции (набор) необходимых таблиц базы данных, представленный в таблице 1.

Таблица 1. – Перечень необходимых таблиц базы данных

Название коллекции	Описание
Users	Список всех пользователей
Candidates	Список всех кандидатов
Offers	Список всех предложений полученных по email
Templates	Список все шаблонов ответов на емейл
Tags	Список всех навыков

Важное достоинство баз данных – возможность хранения сгруппированных данных в разных коллекциях и в разных форматах.

Заключение. В данной статье были рассмотрены основные цели и принципы моделирования базы данных, построение быстрого, удобного API для веб-приложения. Были рассмотрены основные сущности базы данных. Спроектированы адреса для обращения клиентской части веб-приложения к серверной части.